

## **REMARKS**

This paper is submitted in reply to the Office Action dated October 11, 2006, within the three-month period for response. Reconsideration and allowance of all pending claims are respectfully requested.

In the subject Office Action, the specification was objected to by the Examiner. In addition, claims 34-38 were rejected under 35 U.S.C. § 101. Moreover, claims 1, 4-10, 12-19, 22-27, 29-32 and 34-38 were rejected under 35 U.S.C. § 102(e) as being anticipated by U.S. Patent Application Publication No. 2004/0117377 by Moser et al. Furthermore, claims 2-3 and 20-21 were rejected under 35 U.S.C. § 103(a) as being unpatentable over Moser et al. in view of Admitted Prior Art, and claims 11, 28 and 33 were rejected under 35 U.S.C. § 103(a) as being unpatentable over Moser et al. in view of U.S. Patent No. 6,195,658 to Comito et al.

Applicant respectfully traverses the Examiner's rejections to the extent that they are maintained. Applicant has amended the specification, and has amended claims 34 and 37-38. Applicant respectfully submits that no new matter is being added by the above amendments, as the amendments are fully supported in the specification, drawings and claims as originally filed.

Now turning to the subject Office Action, and initially with regard to the specification, the Examiner will note that Applicant has amended the specification as suggested by the Examiner.

Next with regard to the §101 rejections, the Examiner will note that Applicant has amended claim 34 to recite at least one processor, and to clarify that the program code executes on the at least one processor. In addition, claim 37 has been amended to clarify that the computer readable signal bearing medium is "physical," which Applicant understands complies with the Office's current interpretation of §101. Claim 38 has likewise been amended to exclude a transmission medium. Applicant respectfully submits that claims 34-38 as amended are directed to statutory subject matter, and withdrawal of the §101 rejections is respectfully requested.

Now turning to the art-based rejections, and specifically to the rejection of claim 1, this claim generally recites a method of managing a plurality of software development

environments coupled to one another through a cross development environment. The method includes detecting an update made in a first software development environment among the plurality of software development environments, and dynamically modifying a mapping between the first software development environment and a second software development environment among the plurality of software development environments in response to the detected update.

In rejecting claim 1, the Examiner relies on Moser, and in particular, the abstract and paragraphs [0039] and [0082] thereof. However, Moser is not even directed to software development, and does not even mention the concept of a cross development environment. Moser instead is directed to a generic system for enabling heterogeneous clients to access master data stored in a master database, e.g., for use in storing customer or product information. The only references to "development" and variations thereof are found at paragraphs [0003], [0154] and [0173]. The first two paragraphs refer to the development of the software for the heterogeneous system itself, while the last paragraph refers to a possible application of a heterogeneous system in product lifecycle management. Likewise, all of the references to "software" in Moser refer to the design of the heterogeneous system itself. In short, Moser is not directed to a system for developing software. Furthermore, there is no mention of any type of cross-development environment, as is recited in claim 1.

Beyond the fact that Moser is not even directed to software development, Moser also does not disclose the specific steps recited in claim 1. Specifically, claim 1 recites "dynamically modifying a mapping" between first and second development environments in response to an update made in the first development environment. The cited passages in Moser, however, are merely directed to the concept of mapping data between two clients. Paragraph [0039] discusses mapping updates to an object in one client to a format acceptable to a master database, and then further mapping the updates to a format acceptable to a second client. Paragraph [0082] discusses an example of such a mapping process. Claim 1, however, is not merely directed to mapping data between two environments - the claim recites "dynamically modifying a mapping." Put another way,

claim 1 is directed to changing how data in one environment will be mapped to another environment, i.e., changing the mapping itself.

In this regard, Applicant can find no disclosure in Moser directed to modifying a mapping, much less doing so in response to an update made in a software development environment. Moser discusses "mapping rules" that are used to perform the mappings. These mapping rules are stored in the master server and associated with individual clients. However, as discussed at paragraph [0154], the mapping rules are developed by people who develop applications for the clients. There is no disclosure in the reference that the mapping rules may be dynamically modified in response to an update to a software development environment, as would at a minimum be required to anticipate claim 1.

Accordingly, Applicant submits that claim 1 is novel over Moser, and the rejection thereof should be withdrawn.

Applicant also submits that claim 1 is non-obvious over Moser and the other art of record, as there has been no objective evidence presented of a motivation in the art to modify Moser to incorporate the ability to dynamically modify mappings between software development environments. Reconsideration and allowance of claim 1, and of claims 2-15 which depend therefrom, are therefore respectfully requested.

Next with regard to the claim 16, this claim generally recites a method of managing a plurality of software development environments coupled to one another through a cross development environment. The method includes updating a first software development environment among the plurality of software development environments, and notifying the cross development environment of the update made in the first software development environment in response to the update.

Claim 16 is also rejected as being anticipated by Moser. Notably, however, the Examiner does not separately address claim 16, despite the fact that claim 16 does not even recite the same features as claim 1. Nonetheless, as discussed above in connection with claim 1, Moser does not disclose software development environments or a cross development environment. As such, Moser cannot be interpreted to disclose "notifying a cross development environment" of an update made in a first software development

environment. In order to sustain a rejection based upon anticipation, a reference must disclose each and every feature of a claimed invention. Moser falls far short of meeting this burden, and as a result, the §102 rejection of claim 16 should be withdrawn.

Moser furthermore fails to render claim 16 obvious, as there is no disclosure or suggestion in the reference of the desirability of notifying a cross development environment of an update made to a software development environment. Applicant therefore submits that claim 16 is non-obvious over Moser and the other prior art of record. Reconsideration and allowance of claim 16, and of claims 17-18 which depend therefrom, are therefore respectfully requested.

Next with regard to claim 19, this claim generally recites an apparatus that includes a memory configured to store a mapping data structure for use in a cross development environment that couples together a plurality of software development environments, a processor, and program code configured to detect an update made in a first software development environment among the plurality of software development environments, and dynamically modify the mapping data structure to modify a mapping between the first software development environment and a second software development environment among the plurality of software development environments in response to the detected update.

As discussed above in connection with claim 1, Moser does not disclose or suggest dynamically modifying a mapping between first and second software development environments. Claim 19, which recites similar subject matter, is therefore novel and non-obvious over Moser for the same reasons as presented above for claim 1. Reconsideration and allowance of claim 19, and of claims 20-32 which depend therefrom, are therefore respectfully requested.

Next with regard to claim 33, this claim generally recites an apparatus that includes a memory configured to store a mapping data structure for use in a cross development environment that couples together a plurality of software development environments, wherein the mapping data structure includes at least one wildcarded field, a processor, and program code configured to transform a transaction generated by a first software development environment among the plurality of software development

environments into a format compatible with a second software development environment among the plurality of software development environments using the wildcarded field in the mapping data structure.

In rejecting claim 33, the Examiner admits that Moser doesn't disclose wildcarded fields, and instead relies on Comity for allegedly disclosing this feature at col. 5, lines 9-15. However, the combination of Moser and Comity falls far short of disclosing or suggesting each and every feature of claim 33. As noted above, Moser does not disclose or suggest software development environments or a cross development environment. Likewise, Comity is directed to a system for auditing a database, and does not appear to be directed specifically to software development, or to a cross development environment used to coupled together multiple software development environments.

In addition, claim 33 refers to transforming a transaction generated by one software development environment into a format compatible with another such environment using a wildcarded field in a mapping data structure. The Examiner admits that Moser does not disclose this feature. However, Comity similarly fails to disclose such a feature. Comity instead is directed to auditing a database by comparing records from the database with those from a reference database. The cited passage at col. 5 of Comity refers to the fact that a record in a test database may include wildcarded fields. However, these wildcarded fields are not wildcarded fields in a mapping data structure that is used to convert transactions from one format to another.

As such, the combination of Moser and Comity fails to disclose or suggest each and every feature of claim 33. Moreover, there has been no objective evidence presented that would motivate one of ordinary skill in the art to modify Moser to incorporate wildcarded fields in a mapping data structure. Comity, while disclosing wildcarded fields, deals with wildcarded fields in records in a database that is being audited. There is no disclosure or suggestion that such a field could be used in a mapping data structure, much less one used to transform transactions generated by a software development environment.

Applicant therefore respectfully submits that claim 33 is non-obvious over the combination of Moser and Comity. Reconsideration and allowance of claim 33 are therefore respectfully requested.

Next with regard to claim 34, this claim (disregarding the language added to address the Examiner's §101 concerns) generally recites a computer system that includes first and second software development environments coupled to one another by a cross development environment, and program code resident in the first software development environment and configured to notify the cross development environment of an update made in the first software development environment. As discussed above in connection with claim 16, Moser does not disclose or suggest software development environments, a cross development environment, or the notification of a cross development environment of an update made to a software development environment. Accordingly, claim 34 is novel and non-obvious over Moser for the same reasons as claim 16. Reconsideration and allowance of claim 34, and of claims 35-36 which depend therefrom, are therefore respectfully requested.

Next, with regard to claim 37, this claim generally recites a program product that includes program code configured to detect an update made in a first software development environment among a plurality of software development environments coupled to one another by a cross development environment, and dynamically modify a mapping between the first software development environment and a second software development environment among the plurality of software development environments in response to the detected update.

As discussed above in connection with claim 1, Moser does not disclose or suggest dynamically modifying a mapping between first and second software development environments. Claim 37, which recites similar subject matter, is therefore novel and non-obvious over Moser for the same reasons as presented above for claim 1. Reconsideration and allowance of claim 37, and of claim 38 which depends therefrom, are therefore respectfully requested.

As a final matter, Applicant traverses the Examiner's rejections of the dependent claims based upon their dependency on the aforementioned independent claims.

Nonetheless, Applicant does note that a number of these claims recite additional features that further distinguish these claims from the references cited by the Examiner. However, in the interest of prosecutorial economy, these claims will not be addressed separately herein.

In summary, Applicant respectfully submits that all pending claims are novel and non-obvious over the prior art of record. Reconsideration and allowance of all pending claims are therefore respectfully requested. If the Examiner has any questions regarding the foregoing, or which might otherwise further this case onto allowance, the Examiner may contact the undersigned at (513) 241-2324. Moreover, if any other charges or credits are necessary to complete this communication, please apply them to Deposit Account 23-3000.

Respectfully submitted,

January 11, 2007  
Date

/Scott A. Stinebruner/  
Scott A. Stinebruner  
Reg. No. 38,323  
WOOD, HERRON & EVANS, L.L.P.  
2700 Carew Tower  
441 Vine Street  
Cincinnati, Ohio 45202  
Telephone: (513) 241-2324  
Facsimile: (513) 241-6234